

FIG. 2

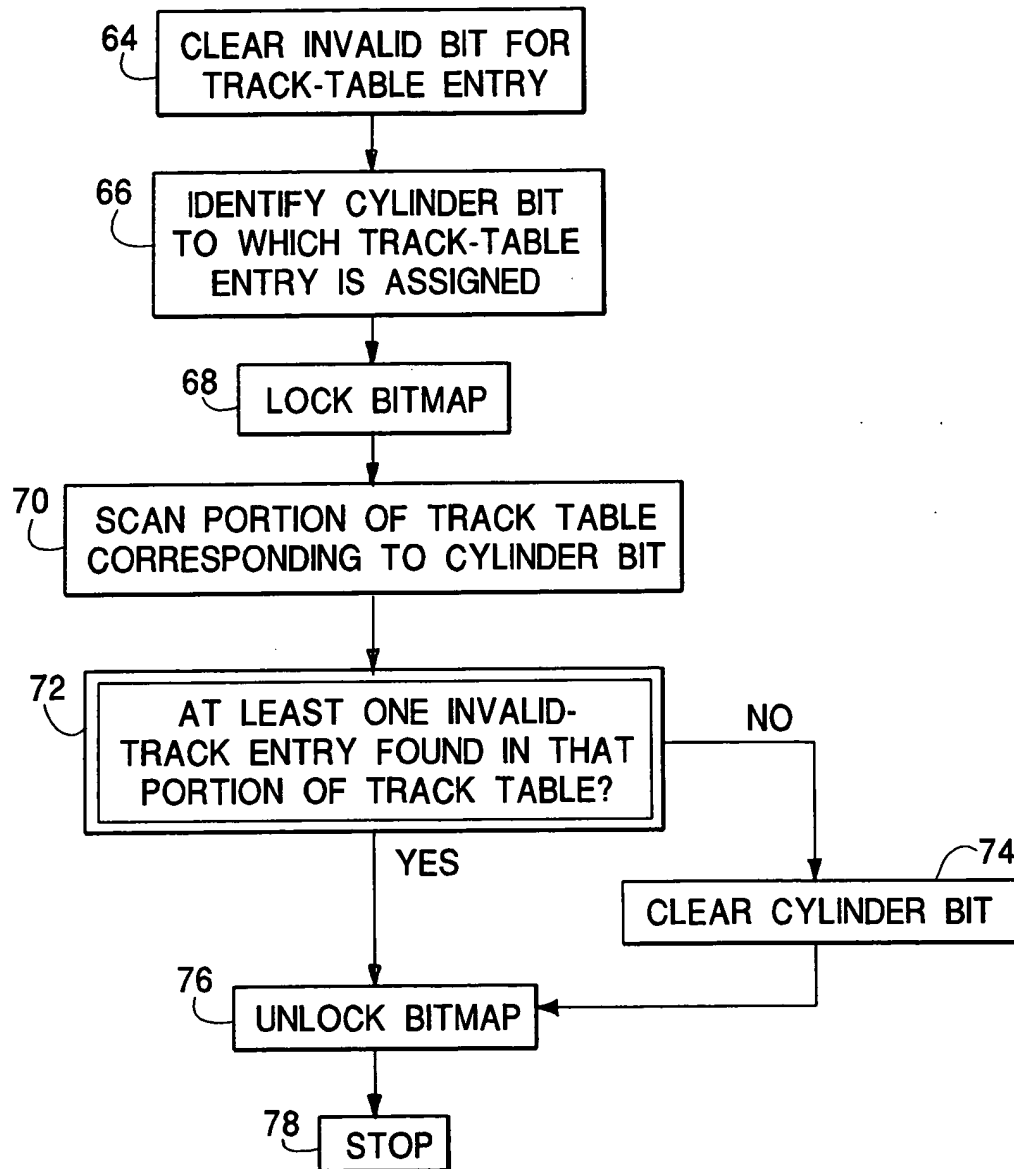


FIG. 3

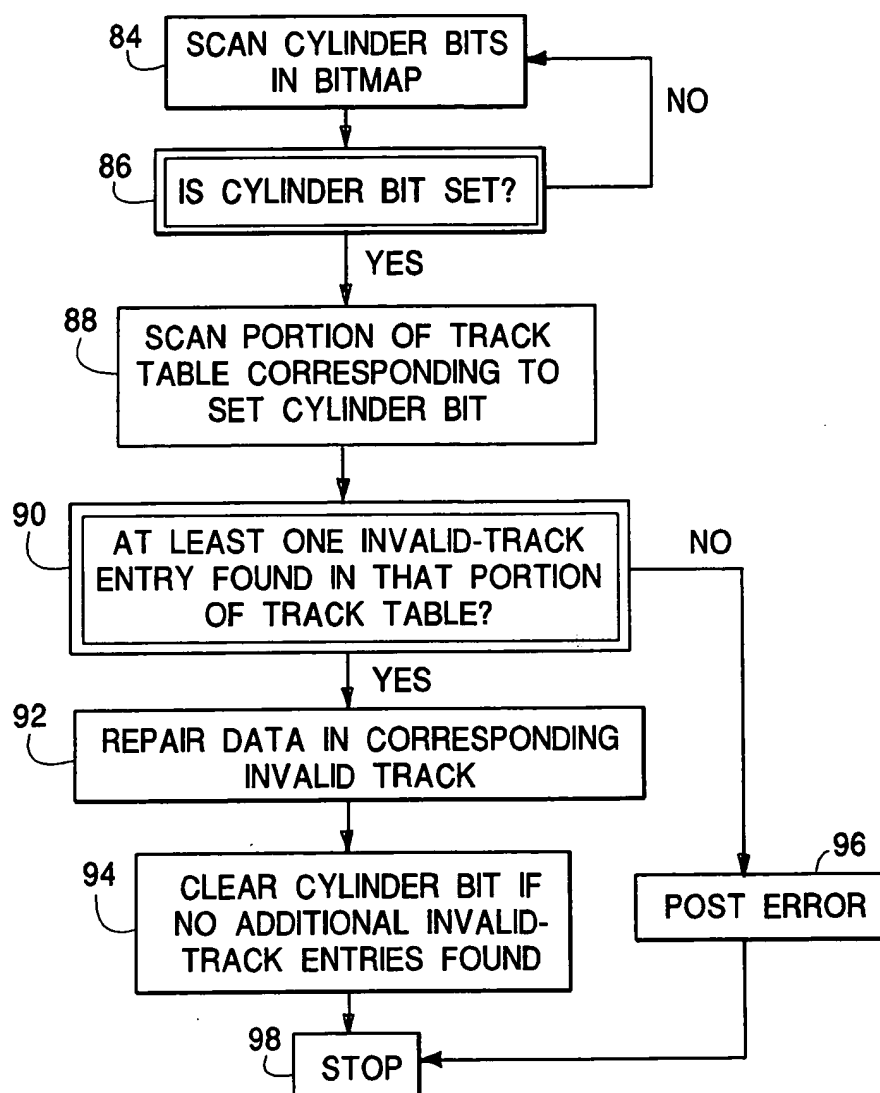
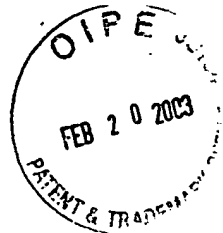


FIG. 4



```
procedure found_invalid_track(T)
100 — [ O = (T/15)/8
      B = remainder(remainder(T/15)/8)
      X = pointer to bitmap
102 — lock bitmap
104 — D = byte at offset 0 from X
      if (bit B of D) = 0 then
106 — set (bit B of D) = 1;
108 — write D at offset 0 from X
      endif
110 — unlock bitmap
```

FIG. 5



```
procedure fixed_invalid_track(T)
112  O = (T/15)/8
113  B = remainder(remainder(T/15)/8)
114  X = pointer to bitmap
115  lock bitmap
116  D = byte at offset 0 from X
117  if (bit B of D) = 1 then
118    count = 15
119    for each track T1 in cylinder containing track T do
120      if T1 is valid then
121        count=count-1
122      endif
123    endfor
124  else
125    trace error
126  endif
127  unlock bitmap
```

FIG. 6



```
procedure find_next_invalid_track(T)
  T = last invalid track
  X = pointer to bitmap
  126 — T1 = remainder(T+1)/n_tracks
  repeat
    128 —  $O = (T1/15)/8$ 
    D=byte at offset O from X
    130 — if D≠0 then
      132 — for each bit B1 in D do
        if bit B1=1 then
          134 — for each track T2 in cylinder containing T1 do
            if T2 is invalid then
              136 — return T2
            endif
          endfor
        endif
      endfor
    138 — T1 = remainder((T1+15)/n_tracks)
    129 — else
      T1=remainder((O*15*8+remainder(T1/15))/n_tracks)
    endif
  until T1=T;
  return -1.
```

FIG. 7